DETAILED ACTION

Continued Examination Under 37 CFR 1.114

A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on 31 August 2009 has been entered.

Response to Arguments

Applicant's arguments with respect to claims 91, 96 and 101 have been considered but are moot in view of the new ground(s) of rejection.

Specifically, deletion of a moved file is obvious and well known as shown below in the rejection. Additionally, Examiner should note that Applicant appears to be focusing on Li's moving of the file, where actually the movement is met by the Lau reference. Li is used to show the obviousness of transcoding a moved file.

Applicant states:

> "As discussed below, all of the claims are in condition for allowance. But if after considering this response, the Examiner does not allow all of the claims, then the Applicant's agent formally requests that the Examiner contact him to schedule and conduct a telephone interview before issuing a subsequent office action."

Examiner acknowledges Applicant's request for an interview. However, Examiner cannot grant an interview on an action in process at this time. If Applicant would like to discuss matters of a completed action on the merits, Examiner requests Applicant contact him formally at 571.272.7516.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

The factual inquiries set forth in *Graham* **v.** *John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966), that are applied for establishing a background for determining obviousness under 35 U.S.C. 103(a) are summarized as follows:

1. Determining the scope and contents of the prior art.
2. Ascertaining the differences between the prior art and the claims at issue.
3. Resolving the level of ordinary skill in the pertinent art.
4. Considering objective evidence present in the application indicating obviousness or nonobviousness.

This application currently names joint inventors. In considering patentability of the claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of

the various claims was commonly owned at the time any inventions covered therein were made absent any evidence to the contrary. Applicant is advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was not commonly owned at the time a later invention was made in order for the examiner to consider the applicability of 35 U.S.C. 103(c) and potential 35 U.S.C. 102(e), (f) or (g) prior art under 35 U.S.C. 103(a).

Claims 91 – 110 are rejected under 35 U.S.C. 103(a) as being unpatentable over Lau (U.S. Patent 6,772,212) in view of Li (U.S. Patent 6,345,279).

Regarding **Claim 91**, Lau discloses:

An electronic device for communication with a user (i.e. computer system; Fig. 1) comprising:

circuitry that includes a processor (processor; Col. 4); and

a set of software instructions that (any of the software to run the system or produce the GUI shown in the figures), when executed by the processor, causes the circuitry to:

display a graphical user interface that includes a library (1202, 1204 and 1206) and that graphically depicts music renderers (Fig. 13 element 1202 which lists the devices that can be communicated with to store tracks) .

Lau does not explicitly disclose that the library is a hierarchical library tree or the music renderers are depicted as nodes.

However, graphical user interfaces for transferring music between portable devices (and non portable devices) are notoriously well known to be shown in a hierarchical format. For example Katz (US 6,356,971) shows a computer with a windowed GUI that includes a main PC connected to remote music devices (Figs. 4A - 4D). Dweyer (US 6,671,567) shows another hierarchical windowed GUI on a computer communicating with a portable audio storage and playback device (Fig 8). Further, the Windows Operating System includes the explorer program which typically shows the various storage devices one can use to store information or manipulate/move.

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify Lau's displays (1202, 1204, 1206) to display as a hierarchical library. One would have been motivated to do so to provide an easier to operate GUI. Hierarchical trees showing files and allowing manipulation have been implemented in systems very widespread (such as MS Windows). As a result user's are familiar with the hierarchical format and thus the device would be easier to use.

The modification further discloses:

wherein the music renderer node identifies a music renderer coupled to the device and (displaying the connected devices in 1202) and includes information about the music renderer (the device connected can be edited, thus indicating that information must be provided to the computer system in order for editing to occur; editing shown in col 13), and the music item node includes an icon identifying a music track stored on the medium of the device (the tracks presented in 1206; also col. 13).

The modification of Lau does not disclose the remaining limitations of claim 91.

In a similar field of endeavor (media transferring between computer devices) Li discloses a system that includes a device that provides capability information to a server device which then is able to transcode the media data to a format acceptable to the device; Fig. 3).

Applying this to the invention disclosed by Lau teaches:

in response to moving the icon from the music item node to the music renderer node (i.e. dragging music files; col. 13), the circuitry

determines whether the format of the corresponding music track is compatible with the corresponding music renderer such that the music renderer can render music from the track (connecting a playback device {such as the one taught by Dweyer example only, many possible} to the computer system, the client profile 310 of the device as modified by Li is provided to the system, which shows the capabilities and resources of the device; col. 6 of Li),

in response to a determination that the format is not compatible with the music renderer, reformats the music track to a format that is compatible with the music renderer; and moves the music track to the music renderer (i.e. Li teaches a content adaptation process 350 uses the profile to select the version of the media that best satisfies the particular client profile, the client, which is the device of Lau, then receives the customized media; col. 6 of Li).

It would have been obvious to one of ordinary skill in the art to apply the teachings of Li to the device taught by Lau. Lau recognizes that multiple devices may be connected to the system. Li recognizes that there exists a wide range of devices

that can provide media content to user. These device contain numerous properties and it would be desirable to optimally match the media to the capabilities of the client device requesting it; cols. 1 and 2.

The combination fails to explicitly disclose:

wherein moving the reformatted music track to the music renderer includes removing the reformatted music track from the storage medium of the device.

However, moving a file from one location to another and deleting the prior file is notoriously well known in the art. As shown in prior actions, the Windows 98 Explorer teaches moving files via dragging and dropping as taught above. In the Windows 98 tutorial, it teaches cutting a file or folder, then pasting it into another directory. Cutting the file and pasting it will remove the original copy. Dragging and dropping can be configured in the same manner. Thus, selecting one directory, dragging the file to the other (in this combination the connected devices 1202) would remove the original file in some configurations. This is precisely how the Windows Explorer file system works. When one drags a file from one folder to another, the original file no longer exists. While Lau is silent on actually removing the original file, it would have been obvious to try as it would have been one of two possible outcomes. Namely, either deleting the file or keep it; each well known in the art.

Furthermore, Doing so would have been nothing more than applying a known technique (i.e. drag and drop with deletion of the original) to a known device (Lau's system w/ attached devices; Fig. 13) ready for improvement to yield predictable results.

Regarding **Claim 92**, in addition to the elements stated above regarding claim 91, the combination further discloses:

an input device comprising at least one of a keyboard, a roller ball, a pen, a stylus, a touch screen, a microphone and a mouse (i.e. inputs to the device such as the pointing device or keyboard).

Regarding **Claim 93**, in addition to the elements stated above regarding claim 91, the combination further discloses:

wherein the hierarchical library tree graphically depicts more than one music renderer node, wherein each music renderer node identifies a respective one of a plurality of music renderers coupled to the device (the devices of 1202 in Lau as displayed in the hierarchical fashion; multiple devices can be connected to the system of Lau; col. 13).

Regarding **Claim 94**, in addition to the elements stated above regarding claim 91, the combination further discloses:

wherein the music renderer nodes represent at least one of a stationary device, a stereo system, a portable device, a Diamond RIO, a RCA Lyra, a portable radio, and a personal display adapter (i.e. a portable device as the hard disk in Lau or the portable device as shown by Dweyer example only, many possible).

Regarding **Claim 95** in addition to the elements stated above regarding claim 92, the combination fails to explicitly disclose in response to moving the icon from the music item node to the music renderer node, the circuitry copies the music track before moving the music track.

However, it is notoriously well known in the art to wait to display a moved item until it is completely copied (i.e. functionality of windows explorer). This is often done to prevent file systems from being corrupted and displaying incorrect information if an error occurs during the transfer process. It would be desirable to add these features to the combination for the same benefits.

Claims 96 – 101 claim the same limitations as claims 91—95 above and are rejected under the same grounds.

Regarding **Claim 102**, in addition to the elements stated above regarding claim 91, the combination further discloses:

wherein the music item node includes more than one icon each identifying a respective one of a plurality of music tracks stored on a storage medium of the device (music tracks 1206 as displayed by a hierarchical tree in the modification).

Regarding **Claim 103**, in addition to the elements stated above regarding claim 91, the combination further discloses:

wherein the hierarchical library tree graphically depicts more than one music item node, wherein each music item node includes an icon identifying a respective one of a plurality of music tracks stored on a storage medium of the device (music tracks 1206 as displayed by a hierarchical tree in the modification).

Regarding **Claim 104**, in addition to the elements stated above regarding claim 91, the combination further discloses:

wherein the circuitry stores the moved music item in the music renderer (storing tracks to the disk via the cartridge; Fig. 1; or portable devices in the examples shown above).

Regarding **Claim 105**, in addition to the elements stated above regarding claim 96, the combination further discloses:

wherein displaying the graphical user interface includes displaying more than one music renderer node, wherein each music renderer node identifies a respective one of a plurality of music renderers coupled to the device (devices 1202 as displayed by a hierarchical tree in the modification).

Regarding **Claim 106**, in addition to the elements stated above regarding claim 96, the combination further discloses:

wherein displaying the graphical user interface includes displaying more than one icon in the music item node, each icon identifying a respective one of a plurality of

music tracks stored in the device (music tracks 1206 as displayed by a hierarchical tree in the modification).

Regarding **Claim 107**, in addition to the elements stated above regarding claim 96, the combination further discloses:

wherein displaying the graphical user interface includes displaying more than one music item node, wherein each music item node includes an icon identifying a respective one of a plurality of music tracks stored in the device (music tracks 1206 as displayed by a hierarchical tree in the modification).

Regarding **Claim 108**, in addition to the elements stated above regarding claim 96, the combination further discloses:

further comprising storing the moved music track in the music renderer (storing tracks to the disk via the cartridge; Fig. 1; or portable devices in the examples shown above).

Regarding **Claim 109**, in addition to the elements stated above regarding claim 101, the combination further discloses:

in response to moving the icon from the music item node to the music renderer node, the program further causes the computer to copy the music track (dragging via the interface; storing tracks to the disk of the computer via the cartridge; Fig. 1; or portable devices in the examples shown above).

Regarding **Claim 110**, in addition to the elements stated above regarding claim 101, the combination further discloses:

in response to moving the icon from the music item node to the music renderer node, the program further causes the computer to store the music track in the music renderer (dragging via the interface; storing tracks to the disk via the cartridge; Fig. 1; or portable devices in the examples shown above).

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to ANDREW C. FLANDERS whose telephone number is (571)272-7516. The examiner can normally be reached on M-F 8:30 - 5:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Curtis Kuntz can be reached on (571) 272-7499. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Andrew C Flanders/
Patent Examiner
Art Unit 2614